

# Inteligentní zobrazování poznámek založené na aktuální poloze a času

Bc. Jan Fogl

Faculty of Informatics and Management

University of Hradec Kralove,

Hradec Kralove, Czech Republic

jan.fogl@uhk.cz

*Tato práce reaguje na čím dál tím větší dostupnost mobilních zařízení v lidské populaci. Do mobilních zařízení se v aktuální době dostává nepřetržité množství senzorů, modulů a výpočetních funkcí, které se dají využít v reálném životě a usnadnit tak lidem práci. Tato práce se konkrétně zabývá využitím GPS modulu a služeb od Googlu pro inteligentní neboli chytré filtrování poznámek, které si uživatel vytvořil v poskytnuté aplikaci. Tato aplikace je vytvořena primárně pro systém android, protože to je v dnešní době nejoblíbenější a nejrozšířenější operační systém. Nejzákladnějším problémem této práce je algoritmus pro vyhledávání poznámek, které jsou svojí lokací vytvořeny co nejbližší k aktuální pozici uživatele. Pro tuto funkcionalitu byla zvolena služba geo fencing z Google Services API. Tato služba umožňuje sdružovat poznámky, které jsou si svojí lokací blízké a poté právě tyto poznámky zobrazit uživateli. Cílem celé práce je tak vytvořit funkční uživatelsky přívětivou aplikaci.*

*Android; Locationing; Location based applications; intelligent application; intelligent filtering*

## I. INTRODUCTION/ÚVOD

### OBSAH

V dnešní době se obrovským tempem rozvíjí svět mobilních zařízení. Tyto mobilní zařízení se postupně dostávají k celé široké veřejnosti a dokáží lidem zprostředkovat zábavu, či pomoci v nejrůznějších pracovních odvětvích. Ruku v ruce s rozvojem mobilních zařízení přicházejí i nejrůznější aplikace založené na přijímání, zpracovávání a vyhodnocování dat z nejrůznějších senzorů, které se dnes stávají standardní výbavou každého mobilního zařízení. Příkladem aplikací využívající tyto systémy mohou být nejrůznější navigační systémy využívající jak systém GPS tak i připojení k internetu, dále systémy rozpoznání gest, nebo systémy, které dokážou lidem šetřit čas i energii.

V našem okolí se vyskytuje velké množství těchto aplikací, některé jsou však nepoužitelné nebo je jejich hranice použitelnosti velmi nízká. Je proto zapotřebí se zamyslet na návrhy těchto aplikací a maximálně je přizpůsobit požadavkům a potřebám co nejširšího spektra uživatelů.

Jako nejrozmumnější a nejčastější využití mobilního zařízení se ukazuje využití aktuální polohy uživatele. Krásným příkladem dobře zpracované myšlenky je upomínková aplikace [1], která umožňuje nastavit upomínku na určitou lokaci a při přiblížení k této lokaci uživateli zobrazí danou upomínku.

Hlavním cílem této práce je vytvoření poznámkové aplikace, která se bude snažit usnadnit práci uživateli, využitím GPS senzoru a připojení na internet. Aplikace bude v době uložení poznámek zaznamenávat aktuální pozici uživatele pomocí GPS a WiFi a následně při prohlížení těchto poznámek bude uživateli nabízet poznámky, které svojí lokací co nejvíce odpovídají aktuální pozici uživatele. Aplikace vychází z myšlenky, že poznámky vytvořené například v zaměstnání si bude uživatel prohlédnout opět v zaměstnání tj. v místě s přibližně stejnou polohou.

Aplikace bude implementována v systému android, který je v dnešní době nejrozšířenější a umožní tak distribuci aplikace co největšímu počtu lidí. Dále systém android nativně obsahuje knihovny pro práci s GPS či datovým připojením.

Získávání polohy bude prováděno primárně pomocí GPS modulu, ale v rámci úspory energie, bude v místech, kde to bude možné využita WiFi a Google API, které umožňuje získat poslední dostupnou polohu uživatele. V článku [1] je popsána možnost, jak získat tuto polohu získat. Existuje široké množství přístupů algoritmů pro seřazení jednotlivých poznámek. Prvním přímočarým způsobem je lokalizace uživatele a následné vypočítání vzdálenosti od jednotlivých uložených poznámek. Algoritmy pro výpočet lokace jsou obsaženy v článku [2]. Dalším způsobem je použití tzv. geo fencingu. Geo fence je oblast vytvořená, v určitém rozmezí okolo daného bodu. Systém android umožňuje monitorovat

všechny vytvořené geo fence a upozorňovat na vstup/výstup z těchto oblastí. Princip geo fencingu je dobře vysvětlen v odborné práci [3].

Princip geo fencingu bude použit v této práci. Pro větší efektivnost bude rozšířen o časový rozměr, to znamená, že řazení prvků bude závislé nejen na aktuálních souřadnicích uživatele, ale i na čase, kdy byla daná poznámka vytvořena. Tento časový rozměr by měl zajistit, aby uživatel získal bez jeho přičinění záznamy, které bude s největší pravděpodobností potřebovat.

## II. PROBLEM DEFINITION/ DEFINICE PROBLÉMU

Jak už bylo zmíněno úkolem této práce je navrhnout inteligentní systém nad platformou operačního systému android pro zobrazování poznámek uživatele v závislosti na jeho aktuální poloze, času, kdy byla daná poznámka vytvořena a času, kdy chceme danou položku zobrazit. Systém android byl zvolen díky své oblíbě a rozšířenosti v celém spektru populace. Nebude proto problém zajistit distribuci aplikace širokému velkému množství lidí.

Hlavní vize celé aplikace je usnadnění správy a vyhledávání již vytvořených poznámek. Tato funkcionalita, by měla uživatelský přívětivá, to znamená, že by uživateli měla ulehčit práci s aplikací a uspořít nemalé množství času, které by strávil ručním vyhledáváním poznámek.

Tato problematika je velmi častým terčem velkého množství vývojářů. Obchody s aplikacemi jsou plné různých inteligentních aplikací, které zobrazují určitá data na bázi GPS souřadnic, nebo časových priorit. Celá tato tematika vedla k vytvoření velkého množství odborných prací a také nemalé množství patentů, které se zabývají jak pozičními systémy, tak cíleným marketingem, to znamená zobrazení dat, které bude uživatel s největší pravděpodobností potřebovat. V následující části práce bude uvedeno, několik možných přístupů ke zjišťování polohy, filtrování položek a zobrazení dat uživateli.

V odborné práci [1] se autoři zabývají zaznamenáváním polohy pro aplikaci spravující upomínky a následné zobrazení upomínek v závislosti na aktuální poloze. Z této práce je možné vyčíst princip ukládání aktuální polohy, kdy je aplikace rozdělena na vnitřní a vnější část. Přičemž ve vnější části je pro získání informací využit GPS modul a v části vnitřní je pro získání polohy použito dostatečně silné WiFi připojení a pomocí služby Google API zjištěna aktuální poloha uživatele. Toto získávání polohy je určitě dobrý a použitelný způsob, proto se v této práci můžeme inspirovat lokalizačním systémem, ale získávání dostupných dat na základě výpočtu vzdálenosti mi přijde poněkud složité a určitě bude lepší navrhnout jiný méně náročný a chytřejší způsob.

V patentové publikaci [4] je zveřejněn velice zajímavý přístup k získávání polohy uživatele. Tento princip je

založen na vysílání radiových frekvencí. Potřebná oblast je rozdělena na několik sub částí, které jsou pokryty daným radiovým signálem. Tyto radiové signály periodicky vysílají informace o aktuální poloze subčástí. Mobilní zařízení tyto informace přijímá a získá z nich svojí aktuální polohu. Tento přístup je velice a zajímavý a netradiční, ale nehodí se pro široké použití, protože nelze dostatečně pokrýt celý svět tímto signálem. Proto se tento způsob hodí pouze do malých prostor například interiérů. Opět bychom zde museli použít princip počítání vzdálenosti, který jak už bylo zmíněno, je poněkud složitý.

Patentový článek [5] se zabývá složitostí zobrazení dat uživateli podle jeho aktuální polohy. Práce se vlastně zabývá cíleným marketingem, kdy podle polohy daného uživatele zobrazí reklamy, nabídky, které by mohly uživatele zajímat. Princip zobrazování je zde založen na tzv. geo fencingu. Oblasti jsou rozdělené na geo fency, které nám umožňují detekovat, kde se daný uživatel nachází. Aplikace se svojí polohou dotáže jestli, a popřípadě ve kterém fencu, se nachází a podle toho zajistí zobrazení reklamy. Tento způsob je podle mě nejvýhodnější přístup, protože zde nemusíme složitě vypočítávat vzdálenosti mezi aktuální polohou a polohou poznámek. Bohužel zde není zakomponován prvek času, o který je vytvářená aplikace obohacena.

Poslední metoda získání aktuálních souřadnic uživatele, která tu bude představena je metoda WSM neboli Weighted Screening Method. Je to metoda získávání pozic čistě z Wifi signálu. Tato metoda je vysvětlena v odborném článku [6]. Tato metoda je založena na databázi pozic jednotlivých Wifi sítí, to znamená, že musí existovat databáze, kde budou uloženy pozice všech známých Wifi sítí. Mobilní zařízení se při snaze zjistit svojí polohu dotáže serveru s databází a oznámí mu aktuálně dostupné Wifi připojení. Server vrátí zpět mobilnímu zařízení souřadnice všech dostupných Wifi sítí. V aplikaci se poté například podle triangulace vypočítá aktuální poloha uživatele. Toto řešení je ovšem výpočetně velice náročné a velmi nespolehlivé, protože ne všechny Wifi sítě jsou uvedeny v databázi.

V této kapitole bylo ukázáno několik směrů, kterými je možné se ubírat. Ani jeden směr není absolutně dokonalý, proto bude nejvhodnější pro vývoj této aplikace zvolit kombinace představených způsobů. Jako nejlepší varianta se jeví kombinace získávání aktuální lokace přes modul GPS případně přes Google API, tento způsob je popsán v práci [1] a následně se pomocí těchto souřadnic dotázat, jestli se uživatel nachází v nějakém fencu a případně uživateli zobrazit všechny poznámky uložené v tomto fencu poefiltrované podle času.

### III. NEW SOLUTION / NOVÉ ŘEŠENÍ

V přechozí kapitole bylo popsáno několik možných přístupů k vytvoření aplikace. V této kapitole bude popsán způsob, kterým bude tato aplikace vytvořena.

Jako řešení pro tuto aplikaci bude zvolena kombinace přístupů již uvedených v předchozí kapitole. Aplikaci můžeme rozdělit na dvě základní části a to získávání aktuálních souřadnic uživatele a samotné vyhledávání nejbližších poznámek. Pro získávání aktuálních souřadnic uživatele bude použit princip, který je popsán v odborné práci [1] a to je používání GPS modulu a v místech kde je dostupné internetové připojení použití Google services API. V další fázi, což jak už bylo zmíněno, je vytvoření a následné získávání a seřizování dostupných poznámek, bude využit princip geo fencingu, který je krásně vysvětlen v odborném článku [3].

Základní princip aplikace tudíž bude ten, že v době vytvoření nové poznámky aktivujeme získání aktuální polohy uživatele. Ve vnitřní struktuře aplikace se ověří, zdali poblíž této lokace existuje nějaký aktivní geo fence, ke kterému by se mohla poznámka přidělit. Pokud tento geo fence nebude existovat, vytvoří se geo fence nový se středem v aktuální pozici a nastaveným určitým radiusem. Při procházení dostupných poznámek se poté zjistí aktuální pozice uživatele, opět se ověří, zdali nespadá do již vytvořeného geo fencu. Pokud v okolí je vytvořen geo fence zobrazí aplikace všechny poznámky obsažené v tomto geo fencu seřazené podle času vytvoření. Pokud ovšem geo fence nebude existovat, značí to to, že poblíž tohoto místa nebyla ještě vytvořena žádná poznámka, proto aplikace vrátí všechny poznámky seřazené podle času jejich vytvoření.

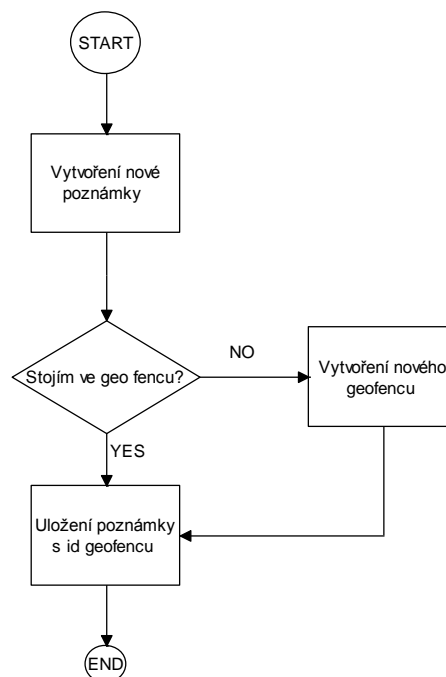
Jak už bylo zmíněno, pro vyhledávání polohy bude použit systém GPS v kombinaci s WiFi. Systém android v sobě integruje přístup k GPS modulu, čímž dokáže ušetřit spoustu práce s vytvářením komunikace se samotným GPS modulem. K GPS modulu se tedy přistupuje pomocí Location manageru, který dokáže zprostředkovat komunikaci mezi aplikací a samotným GPS modulem. Dále je vhodné k Location manageru vytvořit takzvaný Location Listener, který nám umožní snímat aktuální souřadnice na základě několika metod. Tímto popsaným způsobem, získáme aktuální uživateli souřadnice, které pak můžeme poslat dál ke zpracování v další části aplikace. Druhým způsob získání souřadnic bude pomocí připojení k internetu. Existuje několik možností, které již byly popsány v předchozí kapitole, jak získat aktuální souřadnice z WiFi připojení potažmo z připojení k internetu. Pro potřeby této aplikace byl zvolen způsob s využitím Google Services. Google Services je služba od firmy Google, která umožňuje v pravidelných intervalech zaznamenávat aktuální polohu uživatele do cloudu. Avšak tuto službu může mít daný uživatel zakázanou, proto se na informace z Google services nemůžeme vždy spolehnout. V případě potřeby získání

souřadnic jednoduše ověříme, zda je služba aktivní. Pokud aktivní nebude,

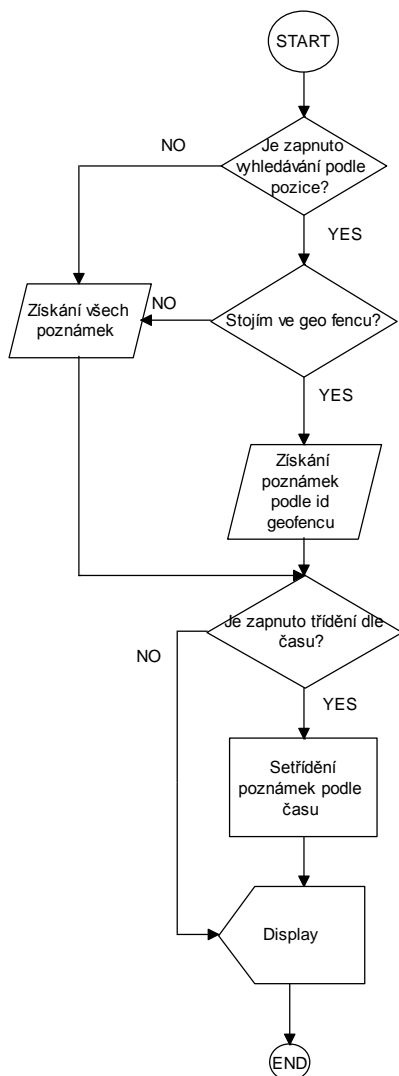
bude se při získávání souřadnic postupovat předchozím způsobem, což je spuštění GPS modulu. Pokud tato služba aktivní bude, získáme pomocí rozhraní poslední známou pozici, kterou budeme považovat za aktuální. Dále opět pošleme souřadnice ke zpracování.

Operační systém android poskytuje službu geo fencing v balíku služeb Google services. Tento balík umožňuje vytvářet, mazat, detekovat a vůbec spravovat všechny věci týkající se geo fencingu to znamená, že přes rozhraní, které Google services je možno obsloužit celou druhou část této poznámkové aplikace. Jak už zde bylo řečeno, při vytváření nové poznámky se získají aktuální souřadnice uživatele a vytvoří se dotaz na rozhraní Google services, jestli v této lokalitě existuje vytvořený geo fence. Pokud ano aplikace si poznámku spáruje s daným geo fencem, pokud ovšem ne, bude opět přes rozhraní Google services vytvořen geo fence nový se zadaným parametrem radius. Při získávání poznámek, které jsou k uživateli nejbližší, se pak opět přes již zmíněné rozhraní zjistí, ve kterém geo fencu se uživatel nachází a přes interní spárování zobrazí všechny poznámky s daného geo fencu.

Průběh vytváření poznámek a jejich následující prohlížení je ukázán v následujících vývojových diagramech.



Obrázek 1 Vývojový diagram vytvoření poznámky



Obrázek 2 Vývojový diagram - Uživatel inicializoval prohledávání poznámek

Poslední částí aplikace je rozšíření o časovou složku. To znamená, že uživatel bude s největší pravděpodobností chtít zobrazit poznámky, které vytvořil v podobnou denní dobu, jako si je chce zobrazit. Kvůli této funkcionalitě projde již jednou vyfiltrované poznámky opětovným filtrováním, které poznámky seřadí podle zadané časové hladiny, to znamená podle absolutní časové vzdálenosti od doby, kdy si poznámky zobrazil.

Aplikace by měla být plně nastavitelná podle uživatelových požadavků, to znamená, že uživatel by měl být schopen ovlivnit v první řadě velikost geo fencu, čímž je myšlena velikost radiusu, podle kterého se budou později vracet vyfiltrované položky. Další nemálo důležitým parametrem ji i nastavení rozpětí času, tj. zdali mají být zobrazeny všechny poznámky z určitého fencu nebo poznámky omezené na časové rozpětí od aktuální doby prohlížení. Toto nastavení by mělo probíhat naprosto

intuitivně bez velkých složitostí a záludností, tak aby aplikace byla k uživateli co nejvíce přátelská.

V této kapitole byl představen návrh řešení, který se snaží zkombinovat, ty nejlepší přístupy z běžně používaných principů a postupů. Konkrétním postupem a implementací se bude zabývat následující kapitola.

#### IV. IMPLEMENTATION / IMPLEMENTACE ŘEŠENÍ

Jak už bylo uvedeno výše, pro implementaci aplikace byla zvolena platforma android. Důvodů ke zvolení této platformy je jistě více, ale hlavním důvodem je všeobecná rozšířenost a dostupnost tohoto systému. Konkrétně byla zvolena verze androidu 4.0.3 a to z důvodů kompatibility i na starších zařízeních.

Nejdříve bylo potřebné navrhnout strukturu aplikace, a systémy, které bude tato aplikace využívat. Na první pohled je jasné, že tato aplikace musí obsahovat databázi. Pro přístup k této databázi byl zvolen Nástroj ORMLite, který značně ulehčuje práci s databází, a dovoluje developerovi zajímat se o důležitější části aplikace. Další technologii nebo jinak řečeno externím systémem je použití Google Services, které v tomto případě slouží k získávání polohy uložené systémem android v cloudu. Struktura aplikace byla zvolena tak, aby byla co nejjednodušší a co nejvíce uživatelsky přívětivá. Aplikace obsahuje jednu hlavní stránku, ze které je možné vytvářet novou poznámku, prohlížet již vytvořené poznámky, zobrazit kalendář uživatele, který je přihlášen v systému android a v neposlední řadě nastavení, kde je možné zapínat, či vypínat systémy třídění poznámek.

Nyní je určitě vhodné projít jednotlivé části aplikace. Aplikace by se primárně dala rozdělit na tři hlavní části, a to na část modelovou, kde jsou obsaženy modelové objekty potřebné ke komunikaci s databází, část pohledovou, kde jsou obsaženy všechny aktivity, které slouží k interakci s uživatelem a část inteligentní, kde děje samotné třídění poznámek podle zadaných kritérií.

V části modelové jsou obsaženy třídy Note a Person a jejich přístupové objekty. Třída Note je objekt, který slouží k uchování informací o poznámce a třída Person je objekt, který uchovává informace o uživateli, kteří jsou s danou poznámkou sdruženi. K těmto třídám samozřejmě náleží jejich přístupové objekty takzvané DAO objekty, neboli data access object. Přístupovou třídu pro objekt Note tvoří tyto následující metody: Metoda save, která slouží k uložení poznámky do databáze, metoda delete, která slouží ke smazání poznámky z databáze, dále jsou tu metody getAll, getById, getName a getByReqId. Tyto metody slouží k získávání poznámek z databáze. Jako poslední metodou je samozřejmě metoda edit, která umožňuje upravovat již vytvořenou poznámku. Přístupová třída pro objekt Person obsahuje následující metody: Metoda save a delete, které slouží k manipulaci se záznamem a metoda getAll, a getById, které slouží ke čtení dat z databáze.

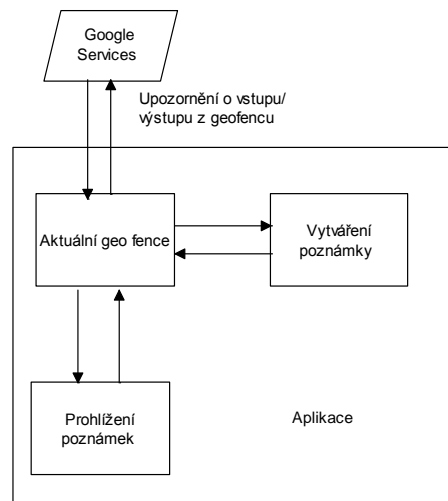
V části pohledové jsou vytvořeny následující třídy. NewNoteActivity, ListNotesActivity, DetailNoteActivity, SettingsActivity a CalendarActivity. Všechny tyto aktivity jsou standartní aktivity systému android bez jakýchkoli úprav. Třída NewNoteActivity obsahuje textové pole, kde je možné přidat název poznámky, textové pole pro vložení těla poznámky, tlačítko pro přidání lidí a tlačítko pro přidání události do kalendáře. Další aktivitou je aktivita k zobrazení detailu jednotlivých poznámek, kde jsou jenom needitovatelné textové pole se zobrazenými hodnotami dané poznámky. Settings aktivita obsahuje pouze dva přepínače, které umožňuje vypínat a zapínat třídění podle polohy a třídění podle času. Poslední aktivita je aktivita, pro zobrazení seznamu již vytvořených poznámek. Aktivita obsahuje standartní ListView komponentu, s poznámkami a jedno textové pole, které slouží k vyhledávání poznámek na základě požadovaného řetězce.

V následujících několika odstavcích bude popsána praktická implementace třídění poznámek.

Třídění na bázi času je jednoduchým rozšířením obvyčejného zobrazování poznámek. K tomuto třídění v aplikaci slouží Třída TimeStrategy. Třída TimeStrategy osahuje metodu sortByTime, která setřídí vložené poznámky podle aktuálního času zobrazení a absolutní hodnoty vzdálenosti od tohoto času. To znamená, že výsledné poznámky nebudou setříděny od nejstarší po nejnovější ale v okruhu podle výchozího času. Třídění je vytvořeno pomocí vlastního komparátoru, který sám poznámky setřídí v závislosti na čase. Do aplikace je tato třída zaintegrována tak, že při zobrazování poznámek je zavolána metoda sortByTime a setříděné poznámky následně zobrazeny.

Třídění na bázi aktuální polohy uživatele je už o něco složitější a navíc vyžaduje připojení k internetu. Princip zobrazování dat tu probíhá následovně. Aplikace si v závislosti na aktuální poloze udržuje buď žádné nebo určité identifikační číslo geo fencu, podle toho jestli se v nějakém nachází nebo ne. Při vytvoření poznámky se podle tohoto identifikačního čísla vytvoří poznámka, to znamená, že pokud uživatele stojí v již vytvořeném geo fencu, použije se toto identifikační číslo u vytvářené poznámky, pokud uživatel v žádném geofencu nestojí, vytvoří se geo fence nový a jeho identifikační číslo se použije pro vytvoření poznámky. Při následném zobrazování jednotlivých poznámek se znovu podle aktuální polohy vybere geo fence, který této poloze odpovídá a poznámky přiřazené do tohoto geo fencu jsou následně zobrazeny. Pokud uživatel v žádném geofencu nestojí, jsou následně zobrazeny všechny dostupné poznámky.

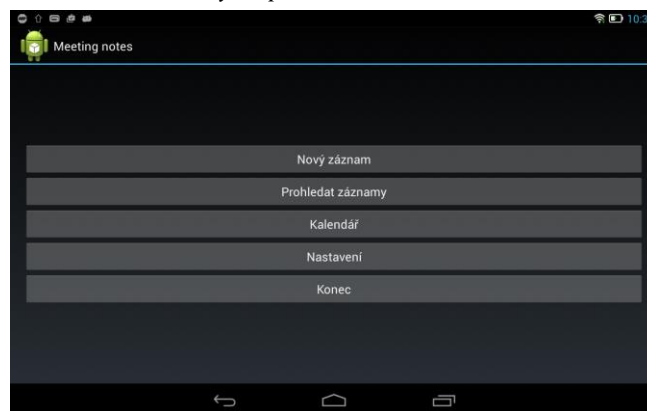
Komunikace jednotlivých částí aplikace s Google services je ilustrována na následujícím obrázku.



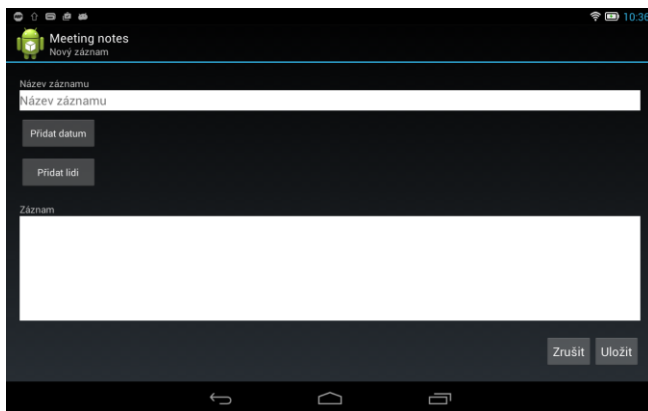
Obrázek 3 Získávání informací o geo fencu

Třídění podle polohy vyžaduje závislosti na GPS modulu a také na internetovém připojení. Implementace pomocí GPS modulu je samozřejmá, ale prakticky se moc nepoužívá, protože je dosti náročná a zkracuje výdrž baterie. Pokud uživateli stačí méně přesná poloha, je samozřejmě lepší využít internetové připojení získat poslední dostupnou polohu ze služby Google services.

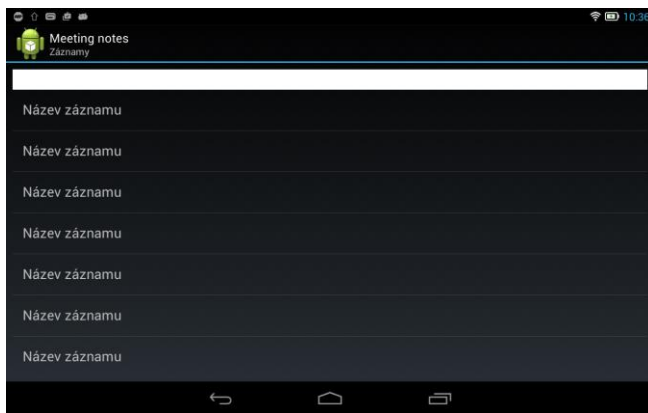
#### A. Pořízené snímky z aplikace



Obrázek 4 Hlavní stránka aplikace



Obrázek 5 Vytváření nové poznámky



Obrázek 6 Prohlédávání poznámek

## V. TESTING OF DEVELOPED APPLICATION / TESTOVÁNÍ VYVINUTÉ APLIKACE - ŘEŠENÍ

Zaměření této kapitoly je zcela jasné, je potřeba stanovit testové scénáře, které umožní následně zhodnotit nově vytvořenou aplikaci.

Testování této aplikace lze pojmut z několika úhlů. Prvním možným přístupem je otestovat rychlost získávání polohy pomocí GPS a porovnat ji s rychlostí získávání polohy pomocí internetu. Druhým úhlem pohledu je testování rychlosti zobrazování jednotlivých poznámek se zapnutým či vypnutým tříděním. Třetím a posledním úhlem pohledu je zjištění, zda má toto třídění vůbec nějaký smysl a hodnotu pro uživatele.

Testování aplikace na rychlost získávání polohy. Test této funkcionality bude prováděn následovně. Do samotného zdrojového kódu aplikace bude vloženo několik časových snímků, které nám umožní zjistit čas, kdy začala aplikace polohu vyhledávat a čas, kdy aplikace vyhledávání dokončila. Sbíráni těchto časových snímků bude samozřejmě provedeno vícekrát a následně bude tento test přehledně zpracován a vyhodnocen.

Druhý test a to test, zdali je možné pozorovat rozdíl v zobrazování poznámek při zapnutém a vypnutém inteligentní třídění bude provedeno obdobně jako u předchozího testu a to z důvodu toho že manuální měření zde není možné, protože časové odchylky budou velmi malé.

Třetí a poslední test má dlouhodobý charakter a je zapotřebí rozšířit tuto aplikaci mezi uživatele. Zároveň je tento test velice subjektivní, protože ho hodnotí každý uživatel sám podle vědomí toho, zda mu aplikace v běžném životě pomáhá nebo ne. Jak už bylo řečeno pro tento test je zapotřebí rozšířit aplikaci mezi uživatele a zároveň vytvořit dotazník, který po určité době umožní uživateli danou aplikaci zhodnotit. Z důvodu tohoto dlouhodobého charakteru nejsou výsledky tohoto testu uvedeny v této práci.

### A. Data nashromážděná testováním

Tabulka 1 Testování času zpro získání polohy - GPS modul

Číslo	Start (ms)	Konec (ms)	Rozdíl (ms)
1	1421693036435	1421693041160	4480
2	1421693202072	1421693221500	19428
3	1421693223857	1421693241568	17711
4	1421693386526	1421693401915	15389
5	1421693451736	1421693462090	10354
6	1421693503735	1421693522153	18418
7	1421693583503	1421693602353	18850
8	1421693640107	1421693642251	2144
9	1421693692179	1421693702424	10254
10	1421693738083	1421693742474	4391
<b>Průměr</b>			12142

Tabulka 2 Testování času pro získání polohy - Google services

Číslo	Start (ms)	Konec (ms)	Rozdíl (ms)
1	1421694174506	1421694174507	1
2	1421694232542	1421694232543	1
3	1421694294023	1421694294023	0
4	1421694351540	1421694351541	1
5	1421694402857	1421694402858	1
6	1421694493421	1421694493422	1
7	1421694554328	1421694554329	1
8	1421694658954	1421694658954	0
9	1421694663251	1421694663252	1
10	1421694712489	1421694712490	1
<b>Průměr</b>			0.8

Tabulka 3 Testování času zobrazení položek - bez třídění

Číslo	Start (ms)	Stop (ms)	Rozdíl (ms)
1	1421695251911	1421695252091	180
2	1421695310783	1421695310896	113
3	1421695351541	1421695351736	195
4	1421695391629	1421695391810	181
5	1421695429307	1421695429484	177

6	1421695461761	1421695461947	186
7	1421695507477	1421695507670	193
8	1421695537080	1421695537264	184
9	1421695579996	1421695580173	177
10	1421695611804	1421695611986	182
<b>Průměr</b>			177

Tabulka 4 Testování času zobrazení položek - tříděné

Číslo	Start (ms)	Stop (ms)	Rozdíl (ms)
1	1421695729034	1421695729214	180
2	1421695782231	1421695782389	158
3	1421695817292	1421695817484	192
4	1421695854215	1421695854392	177
5	1421695887153	1421695887339	186
6	1421695921178	1421695921354	176
7	1421695954818	1421695954990	172
8	1421695991528	1421695991707	179
9	1421696025112	1421696025292	180
10	1421696056612	1421696056788	176
<b>Průměr</b>			177

Z výsledků testů můžeme odvodit tyto závěry. Při testování doby, za kterou je aplikace schopna najít svojí polohu metodou GPS bylo dosaženo průměrně 12142 časových jednotek, oproti tomu při zjišťování polohy pomocí Google services bylo dosaženo průměrně 0,8 časové jednotky. Z tohoto testu vyplývá, že používání zjišťování polohy pomocí Google services je výrazně rychlejší. Druhým testem byl test času zobrazení vytvořených položek. V tomto testu bylo zjištěno, že při třídění položek nebylo dosaženo vyšších času zobrazení než při netřídění, z tohoto testu můžeme usuzovat, že třídění poznámek nemá zásadní vliv na zpomalení aplikace.

Oba tyto testy jsou samozřejmě orientační, protože byly vytvářeny na velmi malém vzorku.

## VI. CONCLUSIONS / ZÁVĚRY

Cílem této práce bylo vytvořit aplikaci na vytváření a správu poznámek uživatele. Tato aplikace měla být rozšířena o inteligentní vyhledávání již vytvořených položek. Jako nejvhodnější řešení bylo vybráno využití geo fencingu pro vyhledávání podle aktuální pozice a řazení poznámek podle aktuálního vyhledávacího času.

Aplikace byla naprogramována a byly na ní provedeny dva základní typy testů. První test měl za úkol zjistit rozdíly ve zjišťování polohy metodou GPS a dotazováním se na službu Google Services. Z testu je patrné, že služba Google Services dokáže dodat polohu mnohem rychleji než GPS, ovšem na úkor přesnosti. V druhém testu byl testován čas přístupu k vytvořeným položkám s a bez třídění. V tomto testu nebyl prokázán výrazný rozdíl mezi oběma přístupy.

Tato aplikace si rozhodně dokáže najít své příznivce, kteří budou s touto funkcionalitou spokojeni. Postupem času s přibývajícím množstvím uživatelů, bude možné provést test spokojenosti a přínosu do běžného života.

## REFERENCES / REFERENCE

- [1] Chi-Yi Lin, Ming-Tze Hung - A location-based personal task reminder for mobile users, Pers Ubiquit Comput (2014) 18:303–314, DOI 10.1007/s00779-013-0646-2
- [2] Hassan Chizari, Timothy Poston, Shukor Abd Razak, Abdul Hanan Abdullah, Shaharuddin Salleh - Local coverage measurement algorithm in GPS-free wireless sensor networks, Ad Hoc Networks 23 (2014) 1–17
- [3] Lindsay Snider, Ann Arbor, - SYSTEMS AND METHODS FOR SHARING OF GEO-TAGGED FILE SEGMENTS BASED ON LOCATION OF A MOBILE DEVICE, Pub. No.: US 2014/0248912 A1, Pub. Date: Sep. 4, 2014
- [4] Avinash Joshi, Saurabh Bhargava, Radhakrishnan Suryanarayanan - SYSTEM AND METHOD OF LOCATING USERS INDOORS, Pub. No.: US 2013/0249672 A1, Pub. Date: Sep. 26, 2013
- [5] Michael J. Zeto, III, David Rippetoe, David Shaw, Renee Mercer, Gilberto Gaxiola, Jr., Ronald Tyrone Williams, Eric Anders Oscar Johansson, - SYSTEM AND METHODS FOR DELIVERING TARGETED MARKETING CONTENT TO MOBILE DEVICE USERS BASED ON GEOLOCATION, Pub. No.: US 2013/0268353 A1 Pub. Date: Oct. 10, 2013
- [6] Hung-Huan Liu, Wei-Hsiang Lo, Chih-Cheng Tsen, Haw-Yun Shin - A WiFi-Based Weighted Screening Method for Indoor Positioning Systems, Wireless Pers Commun (2014) 79:611–627, DOI 10.1007/s11277-014-1876-y
- [7] Lance Obermeyer, Andrew Oliver, Kyle Andrew Oliver, Prashant Shetty, Aditya Rustgi - GEO-FENCE ENTRY AND EXIT NOTIFICATION SYSTEM, Pub. No.: US 2013/0045753 A1, Pub. Date: Feb. 21, 2013